

# MARS

## Multi-core Application Runtime System for the CELL Broadband Engine

SIGGRAPH 2008

Geoff Levand

[geoffrey.levand@am.sony.com](mailto:geoffrey.levand@am.sony.com)

# Introduction to the Cell Broadband Engine

## Heterogeneous Multi-core Processor

### 1 PPE (PowerPC Processor Element)

64 bit PowerPC core

VMX vector instructions

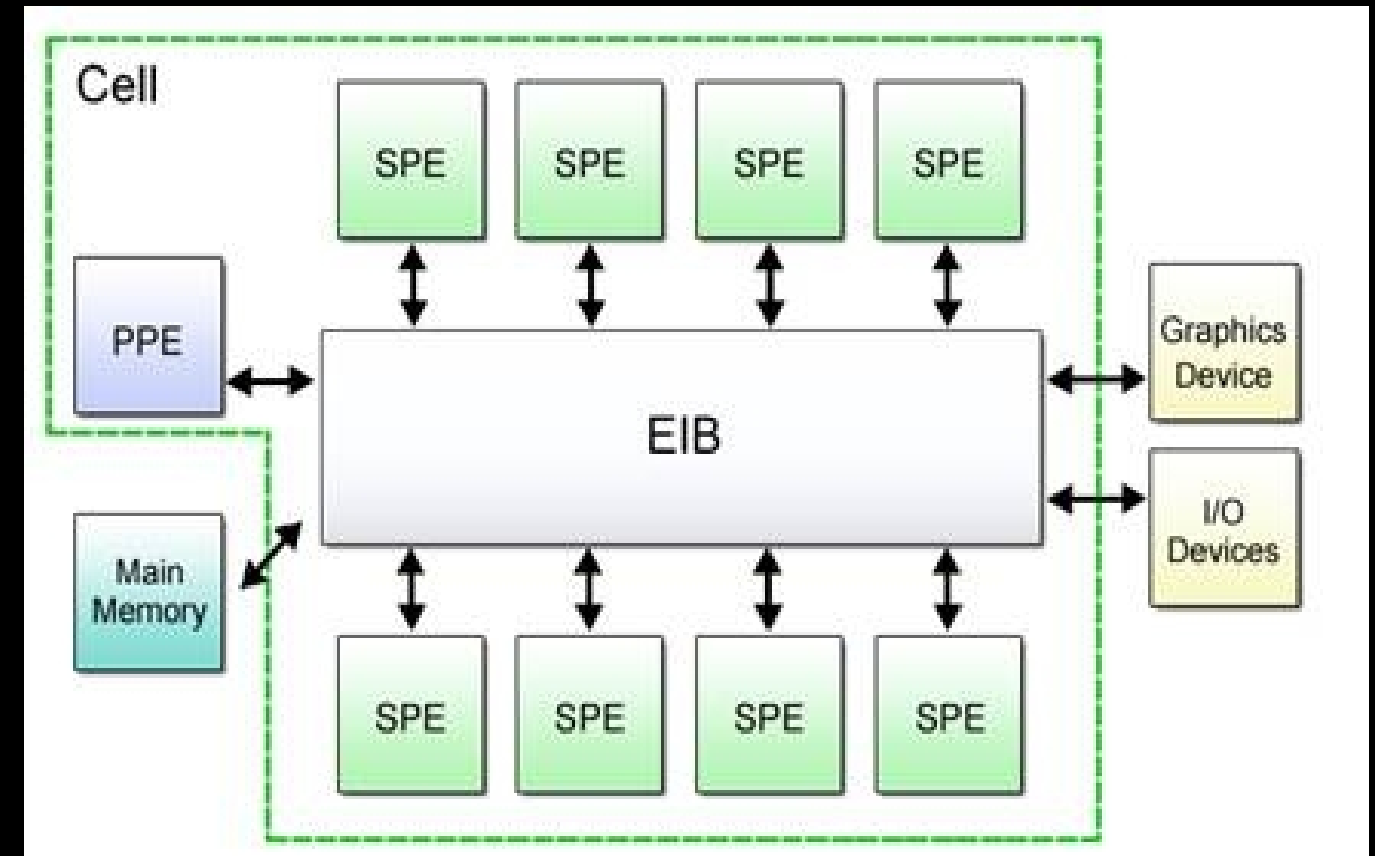
### 8 SPEs (Synergistic Processor Elements)

4-way 32 bit SIMD vector processor

256 KiB Local Store (RAM)

MFC DMA engine

### Very High Speed EBI Interconnect Bus



# Introduction to the Cell Broadband Engine

## Heterogeneous Multi-core Processor

PPE is relatively small

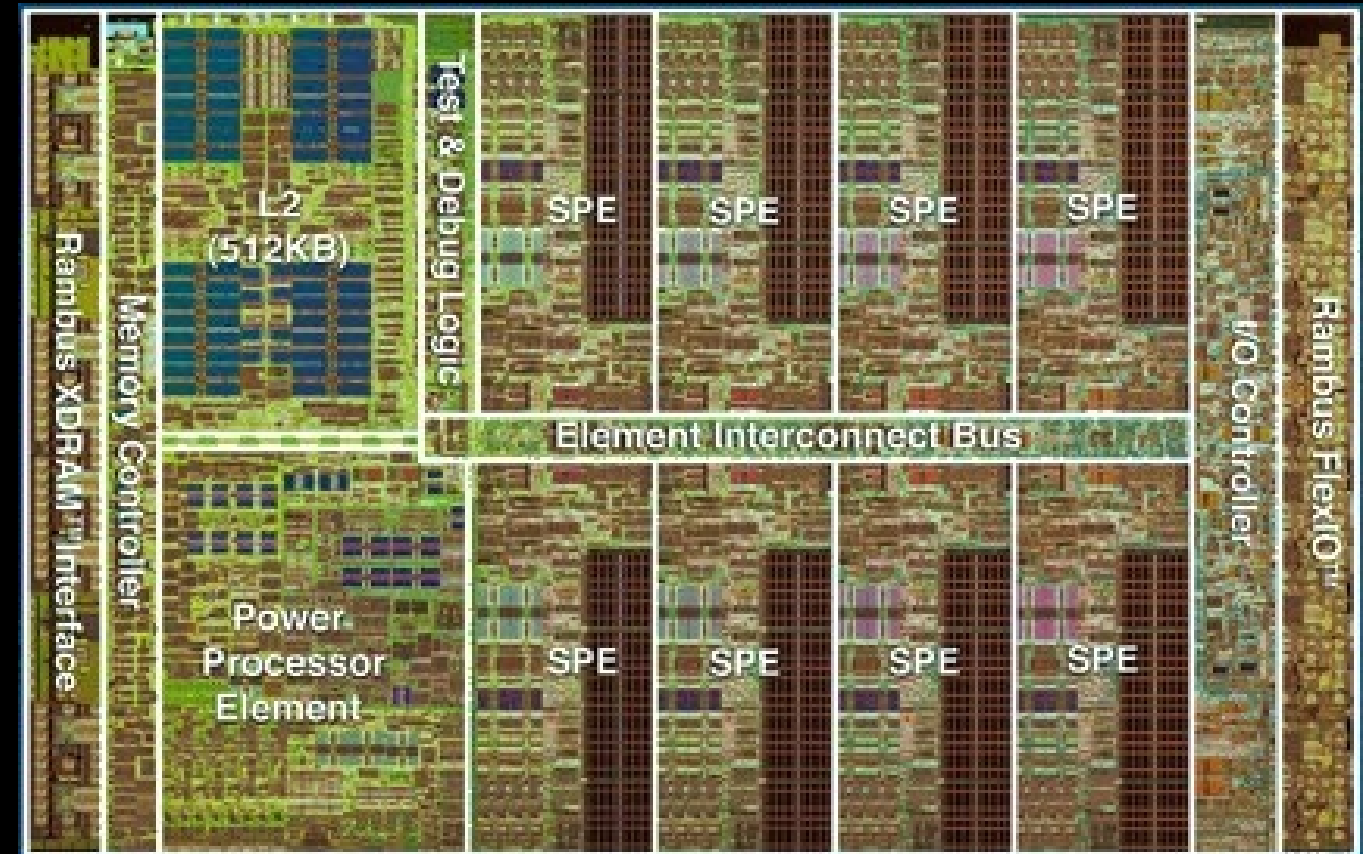
- Leads to efficient energy usage
- Can't perform like bigger CPUs

SPEs take up most of chip

- Excellent number crunchers
- More efficient than general purpose cores
- Easier to program than graphic processors

Must use SPEs to get best advantage

But its fun



# Toolchain for the Cell Broadband Engine

PPE and SPE Have Different Instruction Sets

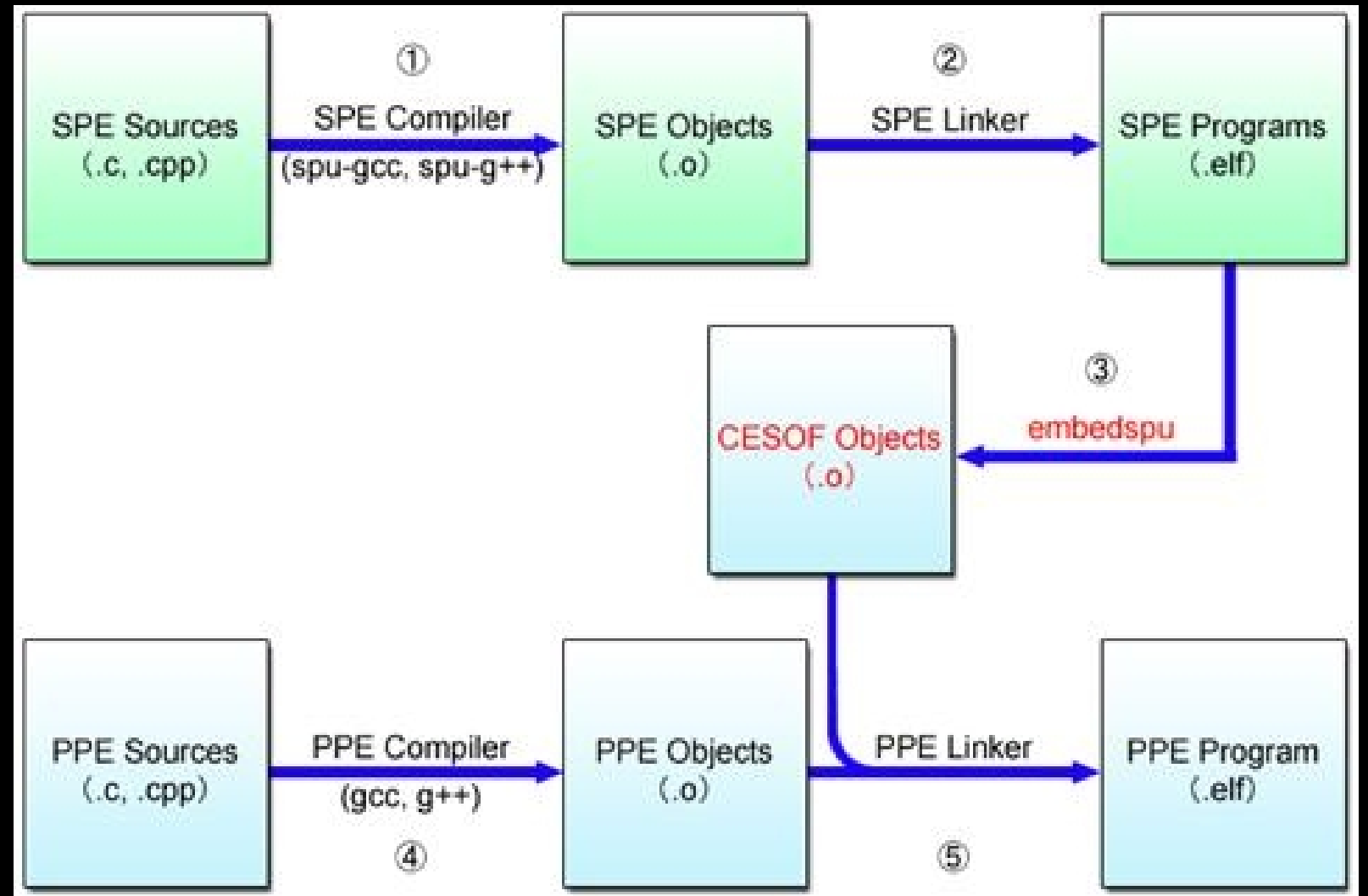
SPE compiler, linker, debugger, etc.

PPE compiler, linker, debugger, etc.

Special PPE embedspu utility converts SPE elf programs to PPE object files

Special Cell 'combined debugger' for debugging programs with PPE and SPE parts

Typical Makefile Logic



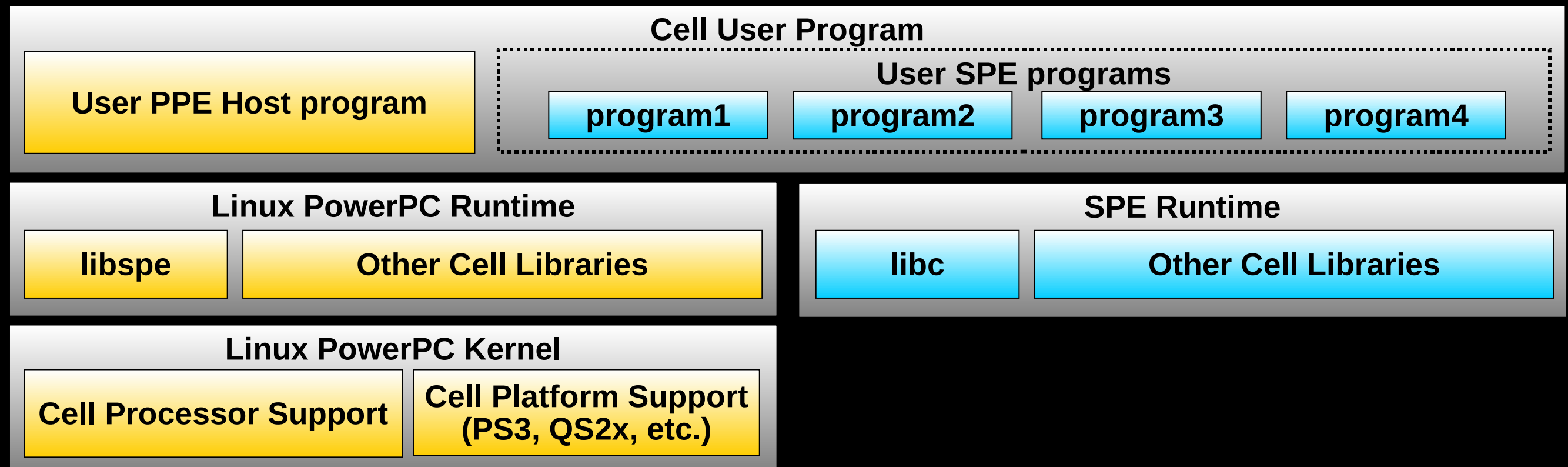
# Linux Support for Cell Broadband Engine

User programs have both PPE and SPE parts

Linux kernel virtualizes physical SPEs and provides low level SPE control interfaces

libspe is used to load and control SPE programs from PPE host programs

SPE libc (newlib) provides a Standard C library for SPE programs



# Linux Kernel SPE Context Scheduler

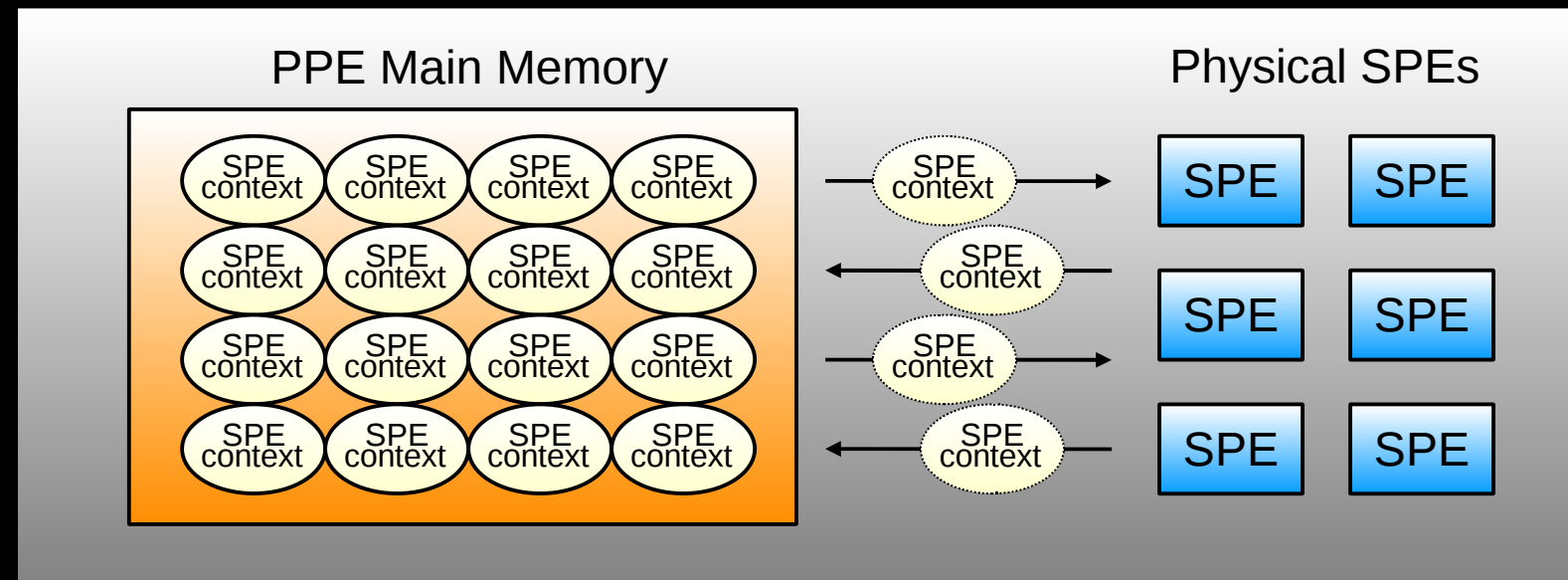
Linux kernel virtualizes physical SPEs

Each SPE program instance managed as an SPE processor context

Linux kernel provides pre-emptive SPE context switching

Entire SPE processor state is stored to, then reloaded from main memory

Potential for high switch overhead when more contexts than physical SPEs



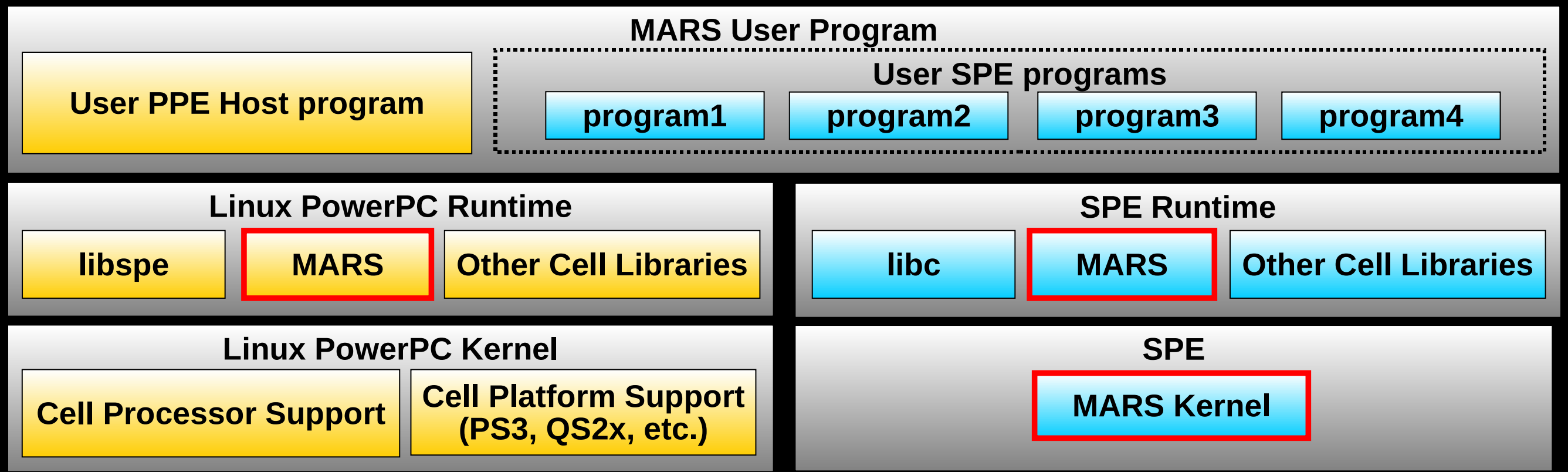
# MARS Components for Cell

MARS PPE programming library

MARS SPE programming library

MARS SPE kernel

MARS GDB debugger extensions



# MARS Terminology

MARS - Multi-core Application Runtime System

Host – host processor (Cell PPE)

MPU – micro-processing unit (Cell SPE)

Host storage – shared memory space (Cell main memory)

MPU storage – MPU local memory space (SPE local store)

Workload – generic scheduling entity of MARS MPU kernel



## **MARS Provides**

MPU-centric runtime for multi-core architectures

Scheduling and management of MARS workloads by MPUs

Lightweight context switching

## **MARS Advantage**

Minimizes runtime load of host processor

Minimizes data exchanged with host processor

Synchronization objects call into scheduler

Simplifies maximizing MPU usage

# MARS SPU Kernel

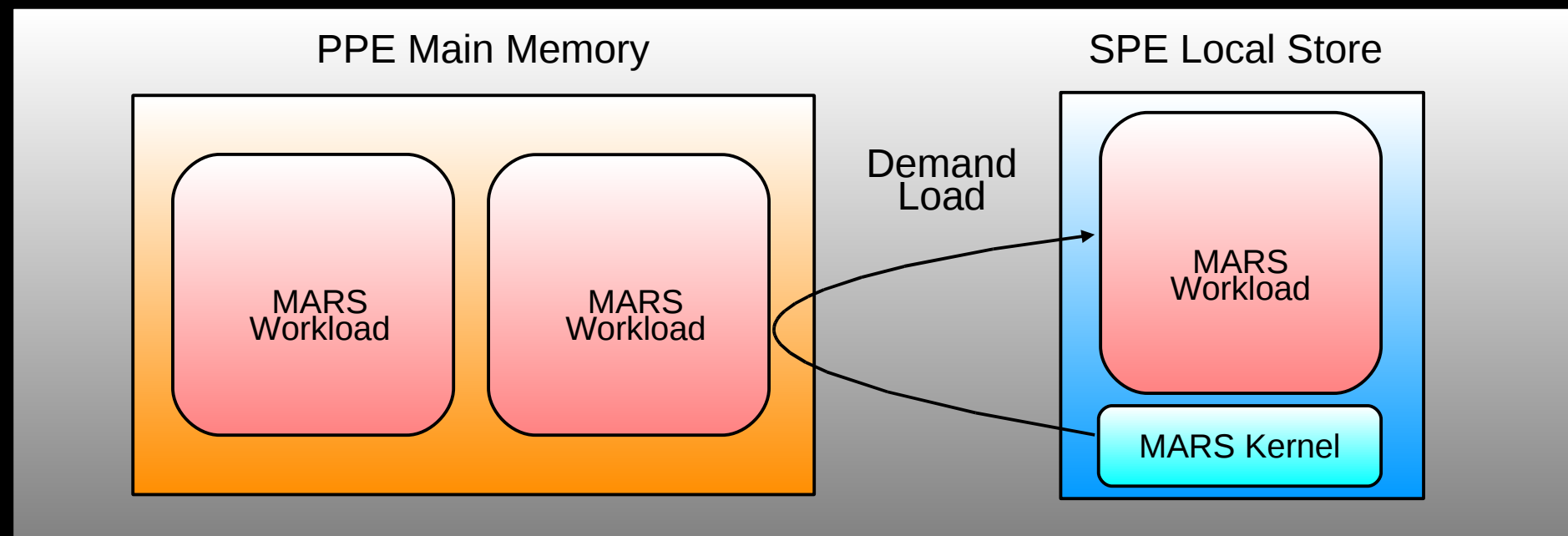
Kernel resident in SPE storage throughout life of MARS session

Kernel demand loads workloads from main memory to SPU memory

Priority-based cooperative scheduling

Content of workload determined by programming model

Task model, Job model, Low level user program, etc.

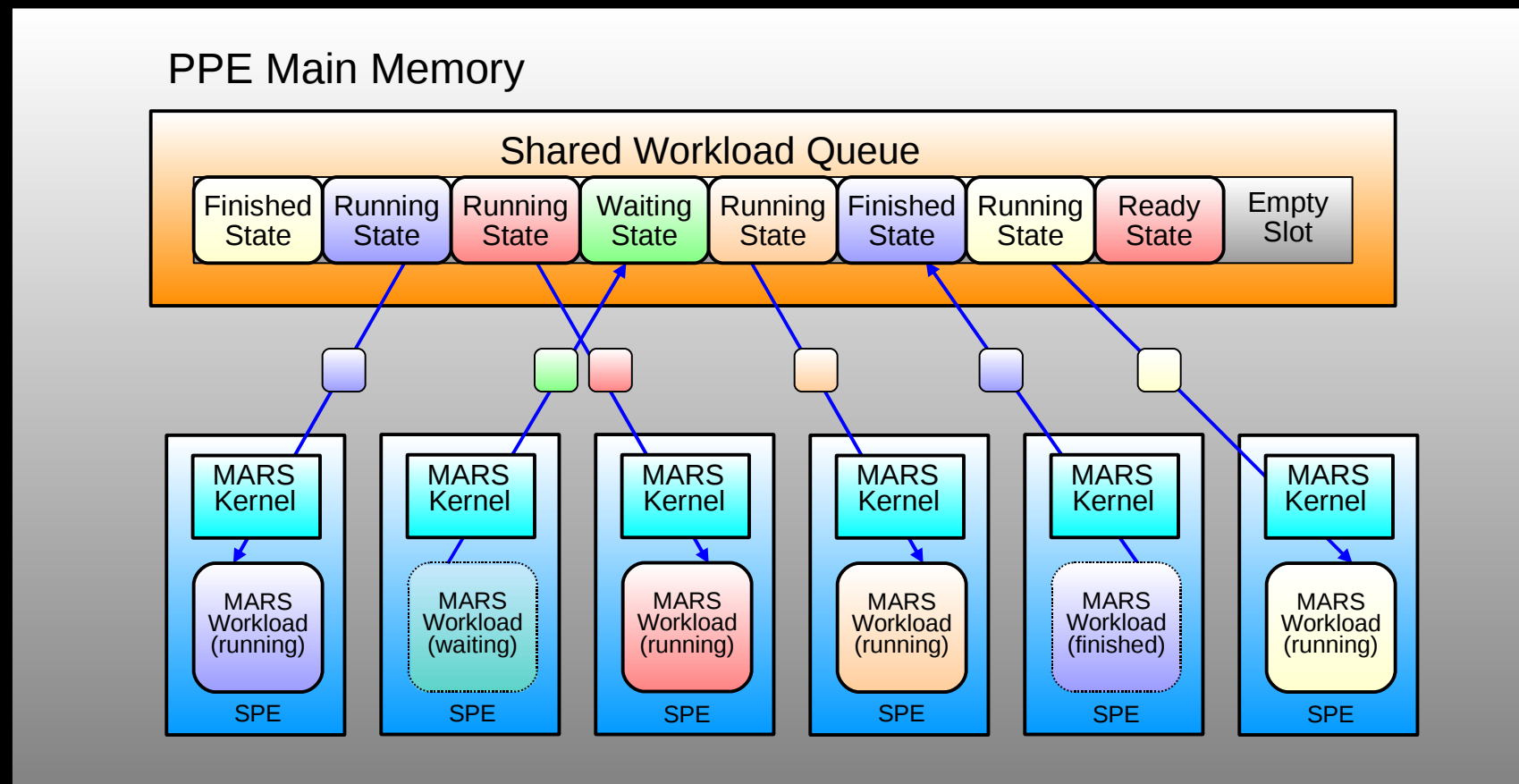


# MARS Workload Queue

Created in main memory and managed through a MARS context

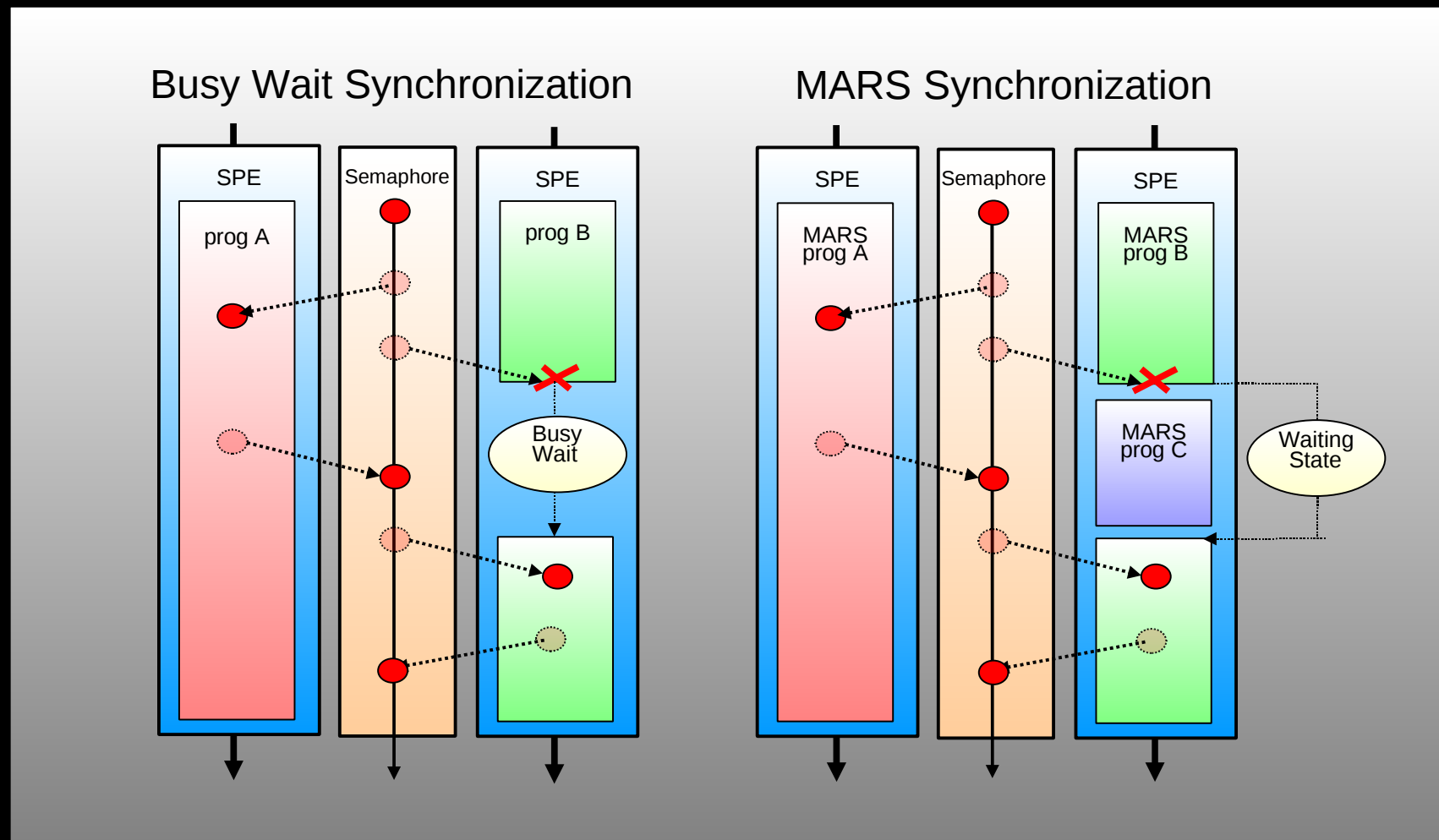
Workload queue atomically accessed by both PPE and SPEs

Workloads in shared queue scheduled by MARS kernels



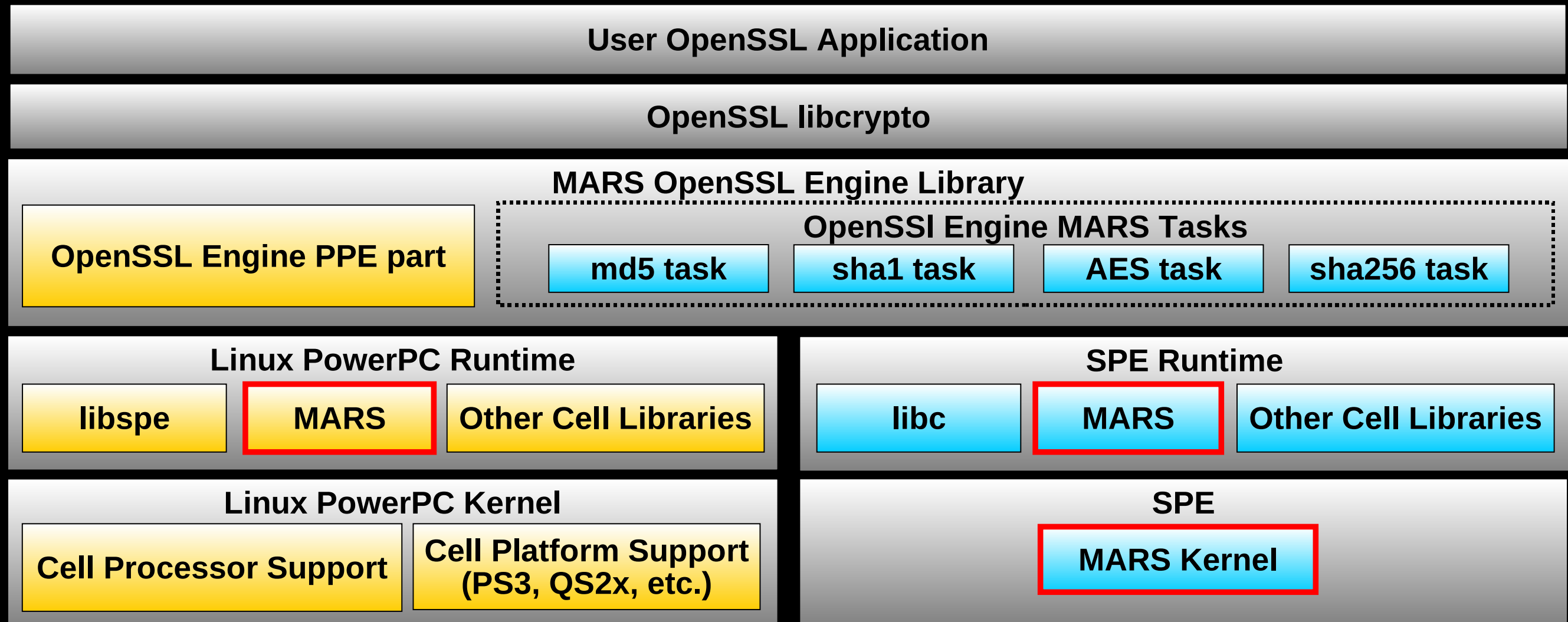
# MARS Synchronization

MARS Synchronization objects call MARS scheduler

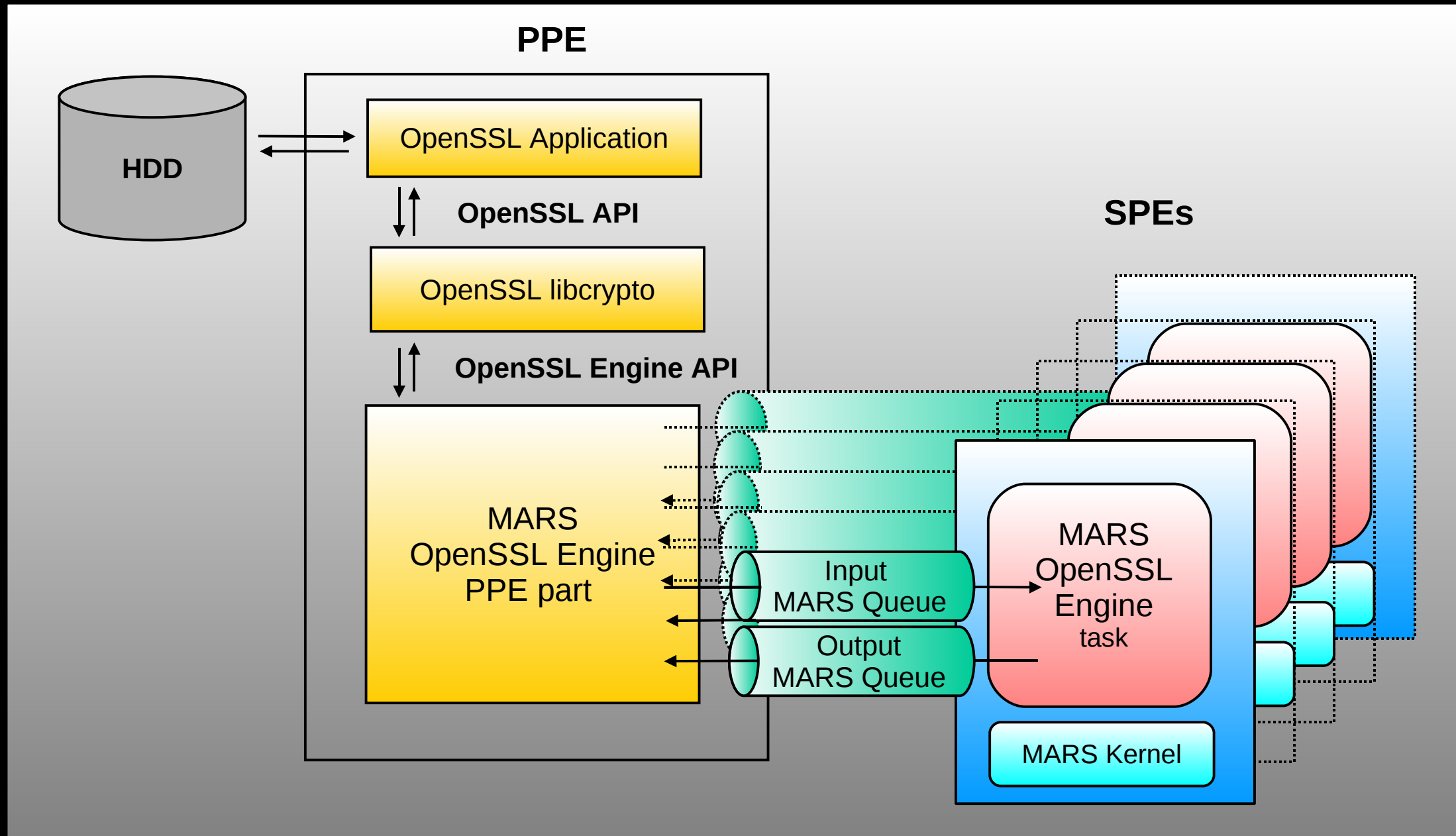


# MARS OpenSSL Programming Example

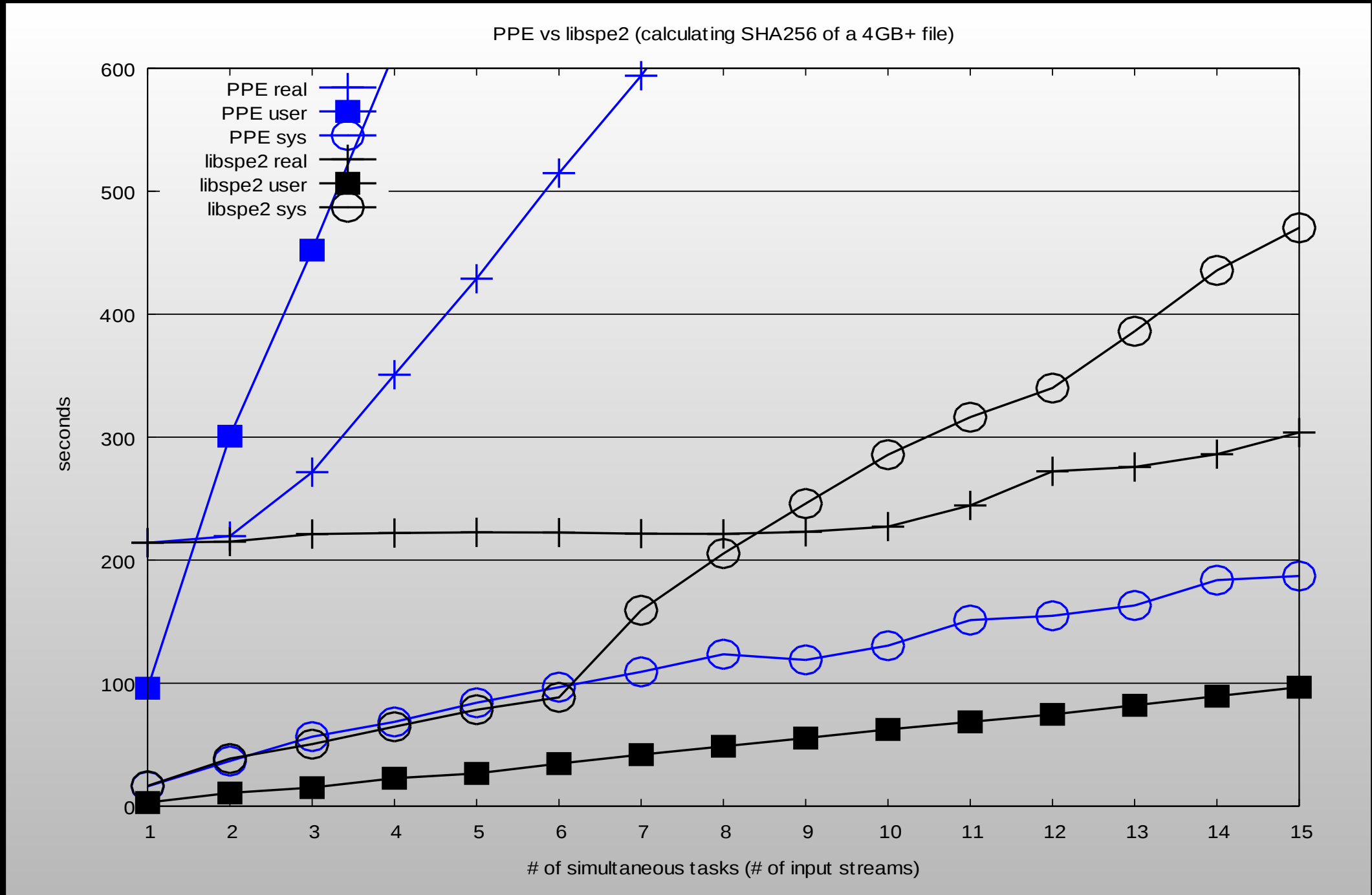
OpenSSL Acceleration Engine implemented using MARS Tasks



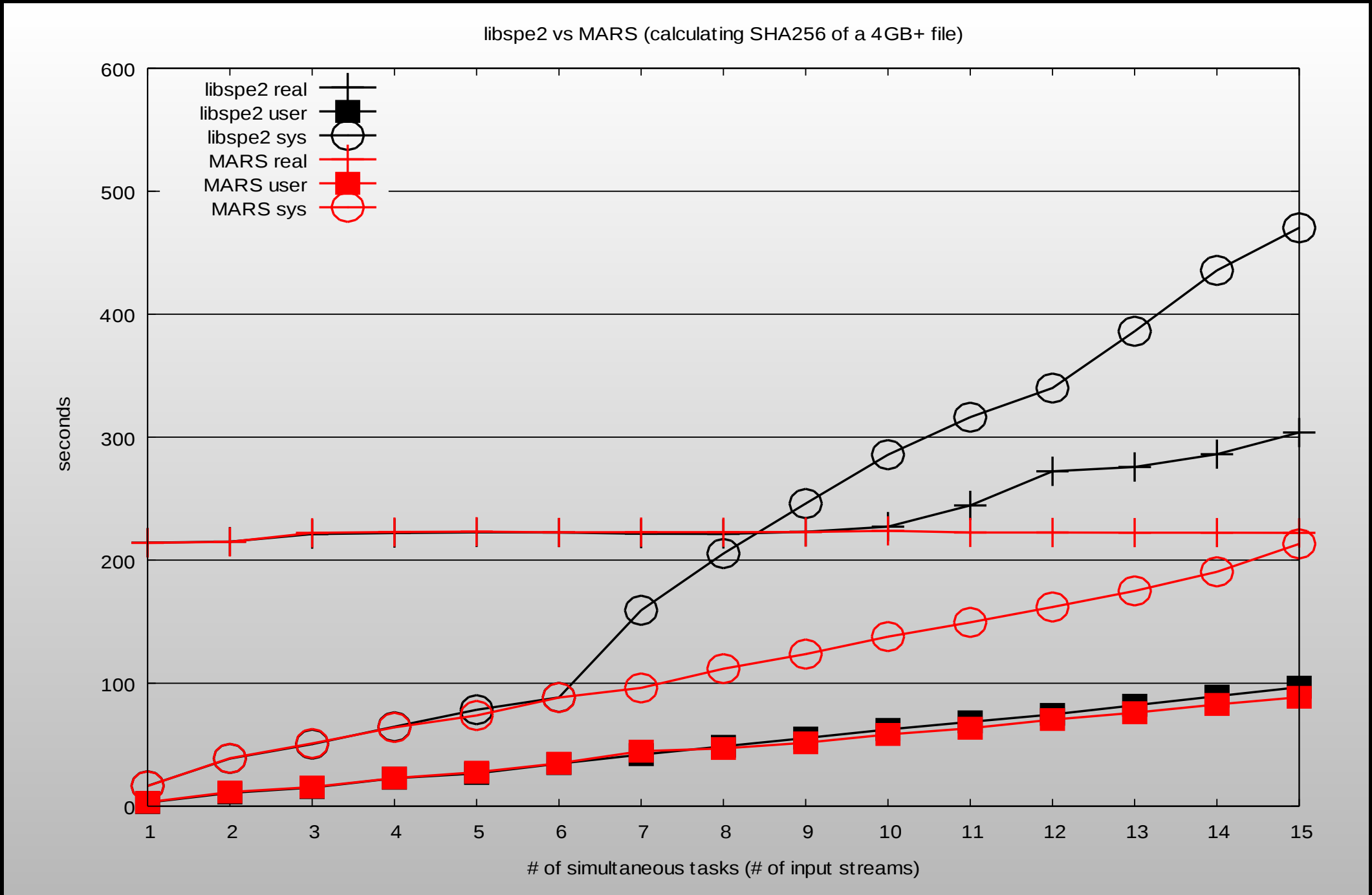
# MARS OpenSSL Example Data Flow



# MARS OpenSSL Results (PPE vs native SPE)



# MARS OpenSSL Results (native SPE vs MARS)





# MARS Current Status (Aug 2008)

Fully functional prototype implementation

Support for Task workload model

Task management and synchronization

Event flag, Barrier, Queue, Semaphore, Signal

# MARS Future Plans

Performance optimizations and feature improvements

MARS Test suite

GDB debugger support for MARS Tasks

Add support for other workload programming models

# More Info on MARS and Cell

MARS and Cell Processor Mailing List Discussions:

[cbe-oss-dev@ozlabs.org](mailto:cbe-oss-dev@ozlabs.org)

MARS and Cell Processor IRC Discussions:

[#cell at irc.freenode.org](irc://irc.freenode.org/#cell)

MARS Releases, Source Code, Samples:

<ftp://ftp.infradead.org/pub/Sony-PS3/mars/>

MARS Development Repositories:

<http://git.infradead.org/ps3/>

Cell Programming Tutorial ( [CellProgrammingPrimer.html](#) ):

<http://www.kernel.org/pub/linux/kernel/people/geoff/cell/ps3-linux-docs/>

Cell Processor Info ( [Cell Broadband Engine Programming Handbook](#) recommended ):

<http://www.ibm.com/developerworks/power/cell/documents.html>

<http://www.bsc.es/projects/deepcomputing/linuxoncell/>

# Thank You

Geoff Levand

[geoffrey.levand@am.sony.com](mailto:geoffrey.levand@am.sony.com)

# Notices

Copyright 2008 Sony Corporation of America

## LICENCE

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

## DISCLAIMER

THIS DOCUMENT IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE DOCUMENT ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS. COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE DOCUMENT OR THE PERFORMANCE OR IMPLEMENTATION OF THE CONTENTS THEREOF.

## TRADEMARK NOTICE

"PLAYSTATION" and "PS3" are registered trademarks of Sony Computer Entertainment Inc.

"Cell Broadband Engine" is a trademark of Sony Computer Entertainment Inc.

Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

Other company, product and service names may be registered trademarks, trademarks or service marks of others.